UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/713,941	11/14/2003	Michael W. Shapiro	03226.348001; SUN040250	8939

32615 7590 09/11/2007

OSHA LIANG L.L.P./SUN
1221 MCKINNEY, SUITE 2800
HOUSTON, TX 77010

EXAMINER	
SMITH, CHENECA	

ART UNIT	PAPER NUMBER
2192	

MAIL DATE	DELIVERY MODE
09/11/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/713,941	SHAPIRO ET AL.	
	Examiner	Art Unit	
	Cheneca P. Smith	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 November 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-21* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-21* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *14 November 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *2/18/2004*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to the application filed on 11/14/2003.

2. Claims 1-21 have been examined.

Claim Rejections - 35 USC § 101

3. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 11-19 and 21 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 11 and 21 both teach a "system" that has been reasonably interpreted as software per se, as it is unclear if the instructions are necessarily in executable form. In addition, claims 11 and 21 fail to claim the system with any hardware and/or computer components cited as to permit the function of the system to be realized. Because the system taught in claims 11 and 21 do not constitute a process, machine, manufacture or a composition of matter, they do not fall within a statutory category of invention and are consequently rejected as nonstatutory – see MPEP 2106.01(I).

Claims 12-19 mirror the deficiencies of claim 11 and are also rejected as non-statutory.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

5. Claims 1,2,9,11;12,16,18-21 are rejected under 35 U.S.C. 102(e) as being anticipated by Bunnell (US Patent Application Publication 2002/0199172 A1).

As to claim 1, Bunnell teaches a method for tracing an instrumented program, comprising:

associating an instrumentation provider with a trace point to provide a probe in the instrumented program (see paragraph [0014], *dynamic instrumentation is performed by inserting, on demand to initiate tracing of events, an instruction at a trace point within the program code as loaded in a memory space of a computer*),

selectively enabling the probe to obtain an enabled probe, wherein enabling the probe includes defining an action to perform when the enabled probe is fired (see paragraph [0034], t*he entry trace data collection control routine 42 preferably functions to determine whether trace data collection is enable and evaluates any constraints on the data to be collected*),

firing the enabled probe during execution of the instrumented program (see paragraph [0034], *provided that trace data is to be collected, the entry trace data collection control routine 42 calls for the execution of an entry-trace process 44 to collect detailed trace data for the current event*),

and performing the action when the enabled probe is fired, wherein the action is performed by a tracing framework (see paragraph [0035], *a call 48 is performed as an initial action of the exit trace data collection control routine 46 to ultimately perform the action of the named function 32*).

As to claim 2, Bunnell teaches the method of claim 1, further comprising receiving a request from a tracing consumer to selectively enable the probe (see paragraph [0034], t*he entry trace data collection control routine 42 preferably functions to determine whether trace data collection is enable and evaluates any constraints on the data to be collected).*

As to claim 9, Bunnell teaches the method of claim 1, wherein firing the enabled probe comprises calling into the tracing framework (see paragraph [0035], *a call 48 is performed as an initial action of the exit trace data collection control routine 46 to ultimately perform the action of the named function 32).*

As to claim 11, Bunnell teaches a system for tracing an instrumented program having a trace point, comprising:

an instrumentation provider configured to associate the trace point to a probe and to enable the probe (see paragraph [0014], *dynamic instrumentation is performed by inserting, on demand to initiate tracing of events, an instruction at a trace point within the program code as loaded in a memory space of a computer*),

a tracing consumer configured to request that the probe be enabled, wherein the request defines an action to perform when the enabled probe is fired (see paragraph [0034], t*he entry trace data collection control routine 42 preferably functions to*

determine whether trace data collection is enable and evaluates any constraints on the data to be collected), and

a tracing framework configured to forward the request to the instrumentation provider and configured to perform the action, if the probe is enabled (see paragraph [0035], *a call 48 is performed as an initial action of the exit trace data collection control routine 46 to ultimately perform the action of the named function 32*).

As to claim 12, Bunnell teaches the system of claim 11, wherein the tracing framework is further configured to create the probe (see paragraph [0035], *another jump instruction is dynamically generated and appended to these preserved instructions to return execution control to the named function 32*).

As to claim 16, Bunnell teaches the system of claim 11, wherein the tracing framework performs the action when the probe is fired (see paragraph [0035], *a call 48 is performed as an initial action of the exit trace data collection control routine 46 to ultimately perform the action of the named function 32*).

As to claim 18, Bunnell teaches the system of claim 11, wherein the tracing framework is configured to register the instrumentation provider (see paragraph [0042], *the trace driver 20 typically in response to a start command 82 issued through the trace collector 22, dynamically instruments the selected named routines in the target program 18*).

As to claim 19, Bunnell teaches The system of claim 11, wherein the tracing framework is configured to unregister the instrumentation provider when the instrumentation provider is unloaded (see paragraph [0044], *the instrumentation code*

dynamically installed is then removed from the target program 18 by selecting each trace point 108 and restoring 110 the entry point instructions previously copied to the corresponding trace buffers 50).

As to claim 20, Bunnell teaches a network system having a plurality of nodes, comprising:

an instrumented program having a trace point (see FIG.1 and associated text, i.e. paragraph [0029]),

an instrumentation provider configured to associate the trace point to a probe and to enable the probe (FIG.1, 20 and associated text, i.e. paragraph [0014]),

a tracing consumer configured to request that the probe be enabled, wherein the request defines an action to perform when the enabled probe is fired (see paragraph [0034], *the entry trace data collection control routine 42 preferably functions to determine whether trace data collection is enable and evaluates any constraints on the data to be collected*) and

a tracing framework configured to forward the request to the instrumentation provider and configured to perform the action, if the probe is enabled, (see paragraph [0035], *a call 48 is performed as an initial action of the exit trace data collection control routine 46 to ultimately perform the action of the named function 32*) wherein the instrumented program resides on any node of the plurality of nodes (see FIG.1, 12 and associated text),

wherein the instrumentation provider resides on any node of the plurality of nodes (see FIG.1, 20 and associated text),

wherein the tracing consumer resides on any node of the plurality of nodes (see FIG.1 and associated text), and

wherein the tracing framework resides on any node of the plurality of nodes (see FIG.1 and associated text).

As to claim 21, Bunnell teaches a system for tracing an instrumented program having a probe, comprising:

a first tracing consumer configured to request that the probe be enabled and perform a first action when fired (see paragraph [0034], t*he entry trace data collection control routine 42 preferably functions to determine whether trace data collection is enable and evaluates any constraints on the data to be collected; provided that trace data is to be collected, the entry trace data collection control routine 42 calls for the execution of an entry-trace process 44 to collect detailed trace data for the current event*) and paragraph [0035], *a call 48 is performed as an initial action of the exit trace data collection control routine 46 to ultimately perform the action of the named function 32*),

a second tracing consumer configured to request that the probe be enabled and perform a second action when fired (see paragraph [0034], t*he entry trace data collection control routine 42 preferably functions to determine whether trace data collection is enable and evaluates any constraints on the data to be collected; provided that trace data is to be collected, the entry trace data collection control routine 42 calls for the execution of an entry-trace process 44 to collect detailed trace data for the current event*) and paragraph [0035], *a call 48 is performed as an initial action of the exit*

trace data collection control routine 46 to ultimately perform the action of the named function 32), and

a tracing framework configured to enable the probe in accordance with the first tracing consumer and the second tracing consumer (see paragraph [0034], *the entry trace data collection control routine 42 preferably functions to determine whether trace data collection is enable and evaluates any constraints on the data to be collected; provided that trace data is to be collected, the entry trace data collection control routine 42 calls for the execution of an entry-trace process 44 to collect detailed trace data for the current event).*

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 4-8, 10, 13,14 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bunnell (US Patent Application Publication 2002/0199172 A1) in view of Boykin et al (US Patent Application Publication 2004/0123279 A1).

As to claim 4, Bunnell teaches the limitations of claim 1, but does not specifically teach disabling the enabled probe if no tracing consumer is requesting the enabled probe. In an analogous art, however, Boykin is cited to teach disabling the enabled probe if no tracing consumer is requesting the enabled probe (see paragraph [0007],

the hooks can manage the execution of the probes, which can be dynamically added or removed from the registry during runtime and/or dynamically enabled or disabled during runtime). It would have been obvious to one having ordinary skill in the art at the time of the invention to combine the teachings of Bunnell and Boykin to gain the ability to dynamically and/or remove probes and to dynamically enable and/or disable probes at runtime, as disclosed by Boykin (see paragraph [0080]).

As to claim 5, Bunnell in view of Boykin, Boykin further teaches removing the probe when the instrumentation provider that provided the probe is unregistered (see paragraph [0007], *the hooks can manage the execution of the probes, which can be dynamically added or removed from the registry during runtime and/or dynamically enabled or disabled during runtime*).

As to claim 6, Bunnell in view of Boykin, Boykin further teaches the method of claim 1, wherein associating the instrumentation provider with the trace point comprises:
determining whether the probe is currently provided at the trace point (see paragraph [0050], *when the injector is notified that a new Java class is being loaded (step 702), it queries the registry to determine whether the newly loaded class needs to be instrumented (step 704)*),
requesting the tracing framework to create the probe if the probe is not currently provided at the trace point (see paragraph [0050], *assuming that the class should be instrumented, the injector then injects hooks at the specified locations (step 708)* thereby *completing the process in the probe injection phase*), and
generating a probe identifier associated with the probe (see paragraph [0046*], the hook*

can determine whether a probe is enabled for its location by querying the registry, e.g. by providing an identifier for the location in which the hook was embedded).

As to claim 7, Bunnell in view of Boykin, Boykin further teaches the method of claim 1, wherein the probe is enabled using the instrumentation provider (see paragraph [0052*], the hook then executes the probes and the probe runtime process is complete).*

As to claim 8, Bunnell in view of Boykin, Boykin further teaches the method of claim 1, wherein selectively enabling the probe comprises:

receiving a request from a tracing consumer to enable the probe (see paragraph [0046], *injector 410 can determine whether a probe is enable for its location by querying the registry; if the registry has an enabled probe, then the hook gets and executes probe 418*),

determining the instrumentation provider that provided the probe (see paragraph [0046], *the hook can determine whether a probe is enabled for its location by querying the probe registry, e.g., by providing an identifier for the location in which the hook was embedded*),

requesting the instrumentation provider that provided the probe to enable the probe (see paragraph [0046], *if the registry has at least one location for the recently loaded class, then the injector proceeds to inject or embed at least on hooks at an indicated method or constructor*) and

enabling the probe by the instrumentation provider to obtain the enabled probe, wherein the enabled probe includes functionality to call into the tracing framework when the

enabled probe is fired (see paragraph [0046], *if the registry has an enabled probe, then the hook gets and executes probe 418*).

As to claim 10, Bunnell in view of Boykin, Boykin further teaches the method of claim 8, wherein the call into the tracing framework comprises a probe identifier associated with the enabled probe (see paragraph [0046*], the hook can determine whether a probe is enabled for its location by querying the registry, e.g. by providing an identifier for the location in which the hook was embedded*).

As to claim 13, Boykin further teaches wherein creating the probe comprises assigning a probe identifier to the probe (see paragraph [0046*], the hook can determine whether a probe is enabled for its location by querying the registry, e.g. by providing an identifier for the location in which the hook was embedded*).

As to claim 14, Boykin further teaches Boykin further teaches wherein associating the instrumentation provider with the trace point comprises:

determining whether the probe is currently provided at the trace point (see paragraph [0050], *when the injector is notified that a new Java class is being loaded (step 702), it queries the registry to determine whether the newly loaded class needs to be instrumented (step 704)*),

requesting the tracing framework to create the probe if the probe is not currently provided at the trace point (see paragraph [0050], *assuming that the class should be instrumented, the injector then injects hooks at the specified locations (step 708)* thereby *completing the process in the probe injection phase*), and

generating a probe identifier associated with the probe (see paragraph [0046*], the hook*

can determine whether a probe is enabled for its location by querying the registry, e.g. by providing an identifier for the location in which the hook was embedded).

As to claim 17, Boykin further teaches wherein the tracing framework is provided with a probe identifier when the probe is fired (see paragraph [0046*], the hook can determine whether a probe is enabled for its location by querying the registry, e.g. by providing an identifier for the location in which the hook was embedded*).

8. Claims 3 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bunnell (US Patent Application Publication 2002/0199172 A1) in view of Inamdar (US Patent Application Publication 2003/0149960 A1).

As to claim 3, Bunnell teaches the limitations of claim 2, but does not specifically teach wherein the request comprises a tuple having a name component, a module component, a function component, and a name component. In an analogous art, however, Inamdar is cited to teach wherein the request comprises a tuple having a name component, a module component, a function component, and a name component (see paragraph [0121], *a probe can be uniquely identified by a library name and a probe name; the probe body can have one or more sections*). It would have been obvious to one having ordinary skill in the art at the time of the invention to combine the teachings of Bunnell and Inamdar for the advantage of gaining a tool that would aid in the development, testing, and analysis of software applications, as disclosed by Inamdar (see paragraph [0011]).

As to claim 15, Bunnell teaches the limitations of claim 11, but does not

specifically teach wherein the request comprises a tuple having a name component, a module component, a function component, and a name component. In an analogous art, however, Inamdar is cited to teach wherein the request comprises a tuple having a name component, a module component, a function component, and a name component (see paragraph [0121], *a probe can be uniquely identified by a library name and a probe name; the probe body can have one or more sections*). It would have been obvious to one having ordinary skill in the art at the time of the invention to combine the teachings of Bunnell and Inamdar for the advantage of gaining a tool that would aid in the development, testing, and analysis of software applications, as disclosed by Inamdar (see paragraph [0011]).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Cheneca P. Smith whose telephone number is (571) 270-1651. The examiner can normally be reached on Monday-Friday 7:00-4:30 EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

CS
8/30/2007

TUAN DAM
SUPERVISORY PATENT EXAMINER